IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 23, 2018, accepted December 30, 2018, date of publication January 18, 2019, date of current version June 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2893485

Multi-Target Tracking With Multiple 2D Range Scanners

HAANJU YOO[1], HYEUN JUN MOON[2], SEUNG-HOON KIM[3], AND SANG-IL CHOI[3], (Member, IEEE)

[1]Department of Computer Science and Engineering, Dankook University, Yongin 16890, South Korea
[2]Department of Architectural Engineering, Dankook University, Yongin 16890, South Korea
[3]Department of Applied Computer Engineering, Dankook University, Yongin 16890, South Korea

Corresponding author: Sang-Il Choi (choisi@dankook.ac.kr)

This work was supported in part by the National Research Foundation of Korea Grant through the Korean Government (MSIT) under Grant 2018R1A2B6001400, and in part by the Korea Institute of Energy Technology Evaluation and Planning from the Ministry of Trade, Industry and Energy, South Korea, through the Human Resources Program in Energy Technology of under Grant 20174030201740.

ABSTRACT We propose a novel efficient online method for tracking performers on stage. Most existing tracking methods have focused on using expensive, high-performing sensors, such as multilayer lidar sensors or high-resolution, short-range radars. Image sensor-based methods are not appropriate for tracking performers on stage because of challenging illumination conditions caused by lighting effects. In this paper, we introduce a robust multi-target tracking method based on the sensor fusing of two-dimensional distance sensors such as single-layer lidar sensors that have a relatively lower cost than the aforementioned types of sensors. In our method, measurements from each sensor are transformed into a reference coordinate system and objects are detected with those transformed measurements. Then, the object detections are used in generating or extending the trajectories of targets by detection-to-trajectory matching. In our experiments, we quantitatively evaluated the proposed method with a newly constructed dataset which consists of two scenarios simulating performances on stage. We collected the scan results of two single-layer lidar sensors and image frames captured by a camera sensor for each scenario. The experimental results show that the proposed method robustly tracks performers in challenging scenarios, in which the performers move abruptly and are densely located.

INDEX TERMS Sensor fusion, multi-target tracking, moving object detection.

I. INTRODUCTION

Estimating the locations of performers on stage is essential for various stage effects. While some commercial systems track performers on stage with various sensors, most existing stage directing systems do not automatically track performers on stage, which causes serious limitations to performance planning. The studies that are most relevant to the tracking of performers on stage are multi-target tracking methods with image cameras [1], [2] and radio frequency (RF) beacons [3]–[6].

Image camera-based multi-target tracking is a classical problem in the computer vision area, and a number of methods have been proposed [1], [2]. Because the cost of image sensors is more competitive than other types of sensors, image sensor-based methods are widely adopted in many applications, especially in the surveillance area. However, the performance of these methods is very sensitive to lighting conditions and the deformation of target objects. Thus, their tracking performance degrades when light changes abruptly or performers take various poses, which are natural conditions of performances on stage.

RF beacon-based tracking methods [3]–[6] are more suitable for tracking on stage because they are not effected by lighting conditions. However, RF systems are too sensitive to the surrounding electromagnetic conditions, so the RF beacon-based method has difficulty in accurately estimating the location of the beacon. Most of them estimate the location of a beacon held by a target with the room-level precision, while step level precision is needed to track performers on stage.

There is another approach to tackle multi-target tracking that is based on a sparse three-dimensional (3D) sensor such as a multilayer lidar sensor [7]–[12]. In this approach, objects and background are represented by clouds of 3D points.

The method accurately tracks targets when they are sparsely distributed. However, when densely distributed targets occlude each other, tracking performance cannot be guaranteed because of the lack of visibility of targets. Although several existing methods adopt additional visual sensors to improve tracking performance [13], [14], they mainly focus on enriching the types of information, not increasing the observability of a sensor network. Moreover, multilayer lidar sensors are too expensive to be widely adopted in practical applications.

In this paper, our primary goal is to develop a simple, cheap and utilizable tracking system that is capable of tracking performers who dance or move abruptly. To this end, we propose a novel and efficient method for tracking multiple performers on stage that resolves occlusion issues with multiple single-layer lidar sensors, which are drastically cheaper than a single multilayer lidar sensor. Kwak *et al.* [15] and Arras *et al.* [16] also proposed a single-layer lidar sensor-based multi-target tracking method. However, they do not utilize the fusing of multiple sensors. On the contrary, we focus on fusing the simultaneous measurements obtained from multiple sensors and detect objects from the fused measurements. Then, our method tracks performers by associating their supposed detections from different scanning times. To our knowledge, there is no available benchmark dataset to address the problem on the multi-target tracking with multiple single-layer lidar sensors. Thus, we generated a new benchmark dataset and evaluated the tracking performance of our method on the dataset. The experiments illustrate that the proposed method robustly tracks performers in challenging scenarios.

To summarize, the contributions of our work include the following.

- Proposing cost-effective tracking system adopting. Our method can be implemented for less than 3,000 USD when the sensor network consists with two 2D lidar sensors and one image sensor.
- Proposing the tracking system that is also scalable. When the number of 2D lidar sensors at different view points, our system has more chance to accurately and robustly track targets. Therefore, the scalability is essential. The computational complexity of the system increases with the number of 2D lidar sensors, thus we can easily increase the number of the sensors with the moderate computational load.
- Constructing a new dataset with multiple 2D lidar sensors and image sensors for the performance evaluation of multi-modal tracking algorithms.

II. RELATED WORKS

In this paper, we propose an efficient and robust multi-target tracking framework. Multi-target tracking is a classical problem which has been studied since the radar sensor was invented. In this field, methods have been studied which estimate the trajectories of targets with measurements from consecutive scans of a sensor. The most primitive way of dealing with multi-target tracking is to apply recursive Bayesian filters such as the Kalman filter [19] to each of targets independently. There are three major issues of multi-target tracking, which have to be resolved for robust tracking: missing measurements, false alarms, and measurement ambiguity. A missing measurement is the absence of a measurement from a target due to sensing error or an occlusion. A false alarm is a measurement obtained from somewhere nothing or a non-target object exists. While missing measurements and false alarms can be resolved easily with consecutive measurements, a measurement ambiguity is much harder to resolve. This is an ambiguity of the ownership of measurements between more than two targets located close to each other. When the targets are located too close, causing occlusion between targets or electrical interfere in obtaining measurements, measurements from each target are merged into one measurement or the measurements of some targets are missed. This degrades the tracking performance of a recursive Bayesian filter, especially when more than two targets have overlapping candidate measurements, because of the uncertainty of their estimations.

To resolve this issue, a joint probabilistic data association filter (JPDAF) [20] and the multiple hypothesis tracking (MHT) framework [21] have been proposed. Both of them track each target with a recursive Bayesian filter such as a Kalman filter [19], but differ in the method of solving the measurement ambiguity. When a recursive Bayesian filter is updated in JPDAF, the probabilities of each measurement about each of the measurement is obtained from each target are calculated by considering the estimated result and the uncertainty of the filter. Then, the current state of the filter is updated with considering the probabilities of all nearby measurements. Therefore, not only one filter uses multiple measurements for its update but also one measurement can be used to update several filters. JPDAF is mathematically concrete and it can mitigate the merging of trajectories compared to the method in which each filter picks up the measurement independently. However, when targets are closely located for a long period, JPDAF also cannot avoid the merging of trajectories.

To prevent the issue of merging of trajectories and recover from the measurement ambiguity, MHT keeps all possible measurement-to-target associations and select the most probable associations when the ambiguity is resolved. Because MHT holds its decision before the ambiguity is resolved, MHT is a kind of deferred method. This approach is called the multiple hypothesis tracking framework because each of propagations of measurement to target associations is called a hypothesis. MHT has the advantage of robust tracking performance compared to other methods even when the target is densely distributed. However, since the number of hypotheses increases exponentially during the measurement ambiguity, it is intractable to find the exact solution due to the extreme computational load.

There are more recent multi-target tracking algorithms based on finite set statistics (FISST) such as the probability hypothesis density (PHD) filter [22], [23],

its extensions [24], [25], and multi-Bernoulli (MB) filter [26]. The methods calculate an approximation of the probability density function about a joint distribution of the unordered target states [18]. These works carefully modeled the motion of targets which successfully capture the dynamics of targets with a single, high-resolution sensor. However, those complicated models are inappropriate for the tracking with a low-resolution sensor network. There are many methods have proposed to track targets with a low-resolution sensor network [27]–[33], but they suffered from scalability issue. To this end, the acoustic sensor-based method [18] which redundantly formulated data association uncertainty and used augmented target states including binary target indicators for statistical independence. Because of the statistical independence, the method drastically reduced the complexity of the tracking model with multiple sensors. Nevertheless, most of those recent tracking algorithms have focused on tracking targets having relatively smooth movements compare to the movements of performers on stage.

Tracking pedestrians with visual sensors is more similar to our tracking environment than the tracking with radar sensors as in the traditional tracking literature. Recently, a number of multi-target tracking methods in computer vision literature have been studied because of a drop in the price of image sensors, the rapid performance increment of hardware and wide distribution of devices with an image sensor, such as cell phones. As aforementioned, a measurement from an image sensor has a much larger data dimension than one from a radar sensor, and it is more difficult to identify a target object in an image than in radar measurements. Thus, the conventional JPDAF of MHT cannot be directly applied to the visual tracking problem. Recently, a rapid increase in the performance of algorithms that detect objects in images [34]–[36] has made it easier to obtain tracking measurements from images. This has led to the proposal of methods that apply MHT to visual tracking problems [37], which has achieved good performances on benchmark dataset. However, their performance depends heavily on the accuracy of object detection, and in the case of tracking on stage, object detection with an image sensor is degraded because of lighting issues such as low exposure and abrupt illumination changes. Moreover, single image sensor-based tracking methods suffer from target occlusions, which frequently occur on stage. Nowadays, there are many visual tracking algorithms that use multiple cameras [2], [38] to overcome occlusion or missing (and also false alarm) issues, but they still cannot resolve the lighting issue.

Since a new type of sensor has become popular due to the development of sensing devices, there have been many attempts to overcome the limitations of a particular type of the sensor by fusing various types of sensors such as lidar sensors and image sensors [13]–[17], [39]. However, the sensors mainly used in these studies are very expensive equipment, such as multilayer lidar, or are difficult to use in a small indoor environment, such as radar. In order to overcome these limitations, methods using RF beacons have been proposed [4]–[6]. However, in the case of RF beacon, it is difficult to accurately predict the position of beacons when electromagnetic noise interferes with the RF signals received. Nevertheless, it is very complicated to compensate for the strength of the signal from a transmitter according to its battery condition and other electromagnetic conditions even if the triangulation method is used. Furthermore, this kind of RF-based method suffers from occlusion when the installed locations of sensors are not high enough and the performers are densely located.

In Table 1, we summarize related works utilizing various types of sensors. The meaning of each column is as followings. The 'multi-domain' column indicates whether the method uses sensors in a different domain. The 'multi-sensor' column indicates whether the method utilizes multiple sensors sharing a common field of view but having different view points. The other columns show which type of sensor is used

TABLE 1. Summary of related works.

Method	Multi-Domain	Multi-Sensor	Type of Sensor (# of sensors)					
			3D Lidar	2D Lidar	RF Beacon	RGB Cam	Depth Cam	Sound
M. Byeon et al. [1]		✓				✓(2≤)		
H. Yoo et al. [2]		✓				✓(2≤)		
K. C. Whitright et al. [3]	✓	✓			✓(N)			✓(N)
K. Lorincz et al. [4]		✓			✓(20)			
D. Zhang et al. [5]		✓			✓(N)			
O. Woodman et al. [6]					✓(N)			
J. Shackleton et al. [7]			✓(1)					
P. Morton et al. [8]			✓(1)					
J. Yan et al. [9]	✓		✓(1)			✓(?)		
A. Dewan et al. [10]			✓(1)					
A. Asvadi et al. [11]	✓	✓	✓(1)			✓(2)		
L. Huang et al. [12]			✓(1)					
B. R. VanVoorst et al. [14]	✓	✓	✓(1)			✓(2)		
K. Kwak et al. [15]	✓	✓		✓(1)		✓(1)		
K. O. Arras et al. [16]				✓(1)				
D. Thomas et al. [17]	✓	✓		✓(1)			✓(1)	
F. Meyer et al. [18]	✓	✓		✓(1)			✓(1)	
Proposed Method		✓		✓(2)				✓(2≤)

in the method. We also give the number of sensors which are used in the method in the parenthesis. A greater than equal mark means that the method utilizes sensors more than a written number. When the exact number is not verified by the literature, we wrote a question mark instead of a number. A number N indicates the number of targets.

In this paper, we propose a method that uses a single-layer or two-dimensional (2D) lidar sensor, which is inexpensive and can be operated in a small space. To increase the observability of our sensor framework, we adopted an additional 2D lidar sensor that has a common field of view with the original sensor. Although a 2D lidar sensor produces measurements that resemble those of radar sensor, tracking performers on stage with a 2D lidar sensor is a significantly different problem from tracking airplanes or ships with a radar sensor because of the different characteristics of the measurements from targets. In the case of the radar tracking problem, measurements from a target seem to a single lump when targets are separated, which is clear to detect each target. However, in the case of lidar tracking on stage, even a single performer can be observed with several point clouds depending on his or her actions. Therefore, applying classical tracking methods based on a radar sensor such as JPDAF and MHT to our problem is not trivial.

III. PROBLEM STATEMENT

In this paper, our goal is to track moving objects on a planar surface with multiple two-dimensional (2D) distance sensors such as 360-degree laser range scanning sensors called lidar, in an online manner. Let us assume that there is a reference 2D coordinate system $\mathbb{W}$ defined on a planar surface that is shared between all sensors. We call this reference 2D coordinate system the world coordinate system. Let's define an arbitrary point measurement by $z_i = (x_i, y_i, s_i, t_i)$ where $(x_i, y_i) \in \mathbb{W}$ are the 2D coordinates or the measurement in the world coordinate system and $s_i \in [1, N_s]$, $t_i \in [1, T]$ are an index of a sensor and a time stamp of obtaining the measurement, respectively. Here, N_s is the total number of sensors and T is the current time stamp. Let $\mathbf{Z}_t$ be a set of all point measurements that are obtained at time T regardless of which sensor they are acquired from:

$$\mathbf{Z}_t = \{z_i | \forall i \text{ s.t. } t_i = t\}. \tag{1}$$

Then, our goal is to estimate the trajectories of K targets $\mathbf{T} = \{\mathcal{T}_k | k = 1, \ldots, K\}$ that are moving on the planar surface with measurements from the targets up to the current scan $\mathbf{Z}_{1:T} = \cup_{t=1}^{T} \mathbf{Z}_t$. Here, the trajectory of the k_{th} target $\mathcal{T}_k$ is defined by the sequence of estimated locations in $\mathbb{W}$ as

$$\mathcal{T}_k = (x_k^{\tau_k^s}, x_k^{\tau_k^s+1}, \cdots, x_k^{\tau_k^e}), \tag{2}$$

where τ_k^s and τ_k^e are the time stamps of starting and ending of observation of the k_{th} target, respectively.

Ideally, the measurements contain exact locations of targets. However, there are many types of noise prevent sensors from producing accurate measurements. To handle these noisy measurements with an algorithm as simple as possible, we adopt assumptions on the characteristics of targets and sensors as follows:

- A target moves smoothly during one scan of a sensor;
- Since the maximum size of a target is fixed, it can be determined whether a cluster of measurements contains measurements from a single target or multiple targets;
- There is no persistent false alarm or missing that last more than seconds;
- A target could not be tracked if it jumps higher or crawls lower than the scanning range of sensors; and
- A target that is fully occluded more than a predefined time interval will be lost in our tracking system.

In the following sections, we describe the overall framework of the proposed method and details on each step of the method. We also show the experimental results of a quantitative evaluation of the method.

IV. SYSTEM OVERVIEW

In this section, we briefly introduce the overall scheme of our method. Fig. 1 is a block diagram of the overall scheme of the proposed method and Fig. 2 shows an example of the output of each step. Our tracking algorithm uses multiple

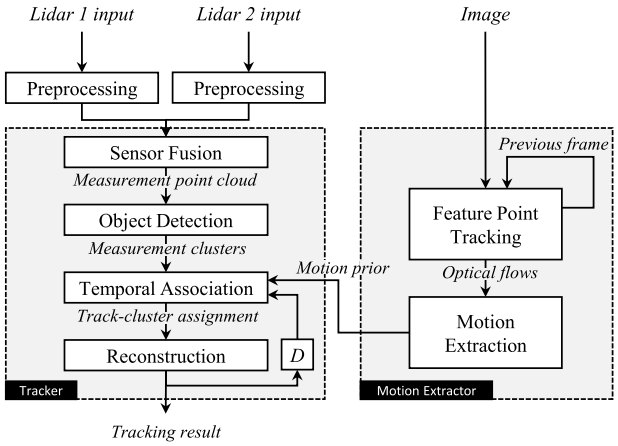
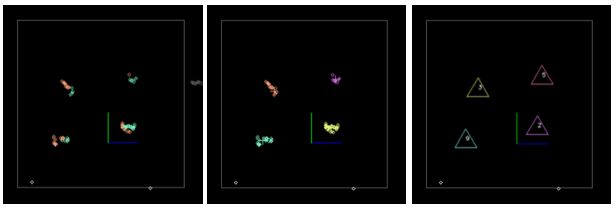


FIGURE 1. Overview of our tracking method with two lidar sensors and a camera. Each white block indicates one step of the proposed method. Each arrow and its label represent data flow. *D* on the feedback loop indicates a delaying function applied to the tracking results.



(a) Sensor fusion (b) Cluster & detection (c) Tracking

FIGURE 2. Example of each step's result (from a rotation scenario in the result section). (a) Merged measurements in the 2D world coordinate system. Measurements from the same sensor are shown in the same color. (b) Objects detected by clustering and detection. (c) The final locations are obtained by temporal association and reconstruction.

2D distance sensors as its input source. Preprocessing is applied to raw measurements from sensors to handle missing measurements or outliers in the raw measurements. See Section V for more details. After preprocessing, the coordinates of sensors that are in the coordinate system of each sensor are transformed into the reference coordinate system, which we call the 2D world coordinate system, which is shared among sensors by a sensor calibration procedure. We describe sensor calibration and sensor fusion in Section VI in detail. To recognize and detect moving objects, we cluster the resulting measurement set of the sensor fusion at time t, $\mathbf{Z}^t = \{z_i | t_i = t\}$, and apply an object detection algorithm as described in Section VII. The detected objects are matched with tracking results until the previous scan by a bi-partite matching algorithm. We call this procedure temporal association. For temporal association, we propose a novel similarity between previous trajectories and the current measurement clusters. For more details about the similarity and temporal association, see Section VIII. After temporal association, the current location of each target is estimated by a reconstruction procedure. In the reconstruction, we estimate the locations of a target when measurements are missed by linear interpolation. Then, we apply a smoothing algorithm to refine the trajectory from noisy measurements. See Section IX for more details about the reconstruction step. As a result of the reconstruction step, the current tracking results, which is the trajectory of each target up to the current scan, are returned as the output of the method and kept and delivered to the temporal association module for matching in the next scan.

V. PREPROCESSING

There are many issues that make a 2D distance sensor fail to obtain a measurement from a target, such as being out of range, diffuse reflection, and device error. Every 2D distance sensor has a valid sensing range, which is the maximum distance ensures that the sensor can measure. When an object is located out of the valid range of a sensor, measurements cannot be obtained from the object because a return signal cannot reach the sensor within the sensing time. Diffuse reflection due to the rugged surface of a target also causes measurements to be missed because the strength of the returning signal is not enough to activate a sensor receiver. When a measurement is missed due to being out of range, it also gives us information that there is no object within the valid range from the sensor. Thus, it is not an actual missing measurement. In contrast, other types of missing measurements are just errors that give the wrong information and show what has to be resolved. Therefore, in this paper, we call the absence of a measurement a missing measurement, except that is caused by being out of range.

To handle measurements that are missed, we apply a classification and refinement procedure to the raw measurements in the preprocessing step. In this step, we classify measurements into reliable measurements and error measurements. Then, we estimate the missing measurement by linear interpolation of its neighboring measurements that are not missed. We depict an input and an output of the preprocessing step as an example in Fig. 3. In the following, we describe our preprocessing step in more detail.

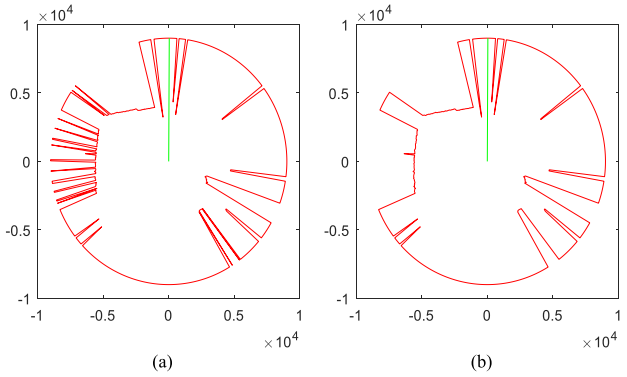


FIGURE 3. Example of preprocessing of lidar measurement. (a) Raw measurements. (b) Results of preprocessing of (a).

A 2D distance sensor such as a single-layer lidar sensor produces its raw measurements with distance values per every unit angle for 360 degrees. Let $z_j^{(k)} = (d_j, \theta_j, t_j)$ be an arbitrary distance measurement obtained by sensor k at time t_j where d_j and θ_j are a distance value and an angle between the reference direction of the sensor and the direction where the measurement came from, respectively. We will discuss the relation between $z^{(k)}$ a sensor level measurement and a measurement in 2D world coordinates in Section VI.

A. MEASUREMENT CLASSIFICATION

As aforementioned, each measurement is either a reliable measurement or an error measurement. in the case of an absence of measurement, we have to determine whether it is a missing measurement or the result of being out of range. In the former case, we regard the measurement as an error measurement and in the latter case, we regard it as a reliable measurement. Let $\mathbf{Z}_s^t$ be the set of all measurements that are obtained by sensor s at time t, i.e.

$$\mathbf{Z}_s^t = \{z_i | s_i = s, t_i = t\} \tag{3}$$

Then, we assume that an arbitrary measurement $z_j^{(k)} \in \mathbf{Z}_s^t$ which has no distance value is an error measurement when the following conditions are satisfied:

1) $\exists z_i$ s.t. $d_i \neq 0, \theta_i - \theta_j \leq \omega_\theta$;
2) $\exists z_k$ s.t. $d_k \neq 0, \theta_k - \theta_i \leq \omega_\theta$; and
3) $\|d_i - d_k\| \leq \epsilon_d$;

where ω_θ and ϵ_d are design parameters, which are threshold values of the range of neighbors and maximum allowable measurement error in distance, respectively. When d_j is determined to be an error measurement, the estimated distance $\hat{d}_j$ is calculated as described in the following section.

B. REFINEMENT

When d_j is determined to be a missing measurement, d_i is estimated with the nearby measurements z_j and z_k as in the

following equation:

$$\hat{d}_j = \frac{(\theta_k - \theta_j) \times d_j + (\theta_j - \theta_i) \times d_k}{\theta_k - \theta_j}. \quad (4)$$

Here, z_i and z_k are the closest measurements that are not missed on the right-side and left-side of z_j, respectively.

Let $\mathbf{Z}$ be an ordered list of measurements obtained by sensor s' at time t' according to their angles. We omit a superscript and sub-script representing a sensor index and a time stamp, for convenience. Next, we describe our preprocessing step in Algorithm 1 in detail. In Algorithm 1 we omit steps for boundary conditions because of the simplification of the algorithm description. For more concrete implementation, see Section X to access our source code.

Algorithm 1 Preprocessing for Lidar Sensor Measurement

Require: Ordered list of concurrent measurements from the same sensor $\mathbf{Z}$, the number of measurements n_Z, search window size ω_θ, maximum measurement error in distance ϵ_d, maximum range of a sensor d_{max}
Ensure: list of refined measurements $\hat{\mathbf{L}}$

$\mathbf{Z}' \leftarrow (z_1, z_2, \cdots, z_{n_Z}, z_{(n_z+1)}, \cdots, z_{(n_z+\omega_\theta/2)})$
$\hat{\mathbf{Z}} \leftarrow \mathbf{Z}'$
while $\hat{\mathbf{Z}}$ is updated **do**
  for z_i in $\mathbf{Z}'_{1:n_z}$ **do**
    if $d_i = 0$ **then**
      $p \leftarrow$ index of nearest preceding non-zero measurement
      if $i - p > 0.5 \times \omega_\theta$ **then**
        continue
      end if
      $f \leftarrow$ index of nearest following non-zero measurement
      if $f - i > 0.5 \times \omega_\theta$ **then**
        continue
      end if
      if $|d_p - d_f| \leq \epsilon_d$ **then**
        $\hat{d}_i \leftarrow$ estimated value by eq. 4
      end if
    end if
  end for
end while
return $\mathbf{Z}'_{1:n_z}$

VI. SENSOR FUSION

We perform sensor fusion, which transforms the measurements taken from multiple multi-domain sensors into a common coordinate system, and then develop the subsequent steps. In this section, we describe how to transform measurements from multiple lidar sensors and image sensors into a 2D world coordinate system $\mathbb{W}$, a common absolute coordinate system shared between all sensors.

A. LIDAR SCAN TO WORLD COORDINATE SYSTEM

In order to fuse the information from multiple lidar sensors and image sensors, a calibration process for identifying corresponding locations in coordinate systems of each sensors is required. Each lidar sensor has 2D polar coordinates as its own coordinate system for representing measurements. In the sensor fusion step, we transform the polar coordinates of a measurement into Euclidean coordinates. Then, the Euclidean coordinates of the measurement are transformed into 2D world coordinates by multiplying the homography matrix H_{s_i} which can be found by the coordinate calibration as the way described in the following paragraphs.

The homography relation is widely used when projecting an image onto a specific flat surface, and it can be used to project the scanning plane of each lidar sensor onto a surface in the 2D world coordinate system, such as the floor of a stage. The final coordinates are obtained by dividing the last element of the resulting homogeneous coordinates which adjusts the scale of coordinates. Let an arbitrary measurement z_i has (d_i, θ_i) as its coordinates in the sensor space, and the 2D world coordinates of z_i, l_i, can be driven by

$$\begin{bmatrix} X \\ Y \\ w \end{bmatrix} = H_{s_i} \times \begin{bmatrix} d_i \sin\theta_i \\ d_i \cos\theta_i \\ 1 \end{bmatrix} \Longrightarrow l_i = \begin{bmatrix} X/w \\ Y/w \end{bmatrix}. \quad (5)$$

In order to obtain a proper homography matrix, we need more than four coordinates in the sensor coordinate system and their corresponding coordinates in the 2D world coordinate system. In the construction of our dataset for experiments, we used a structure that is depicted in the right-most figure in Fig. 4, to extract corresponding points between a lidar sensor coordinate system and the 2D world coordinate system. Because the size of the structure is known, we can figure out the absolute distance between points in each sensor coordinate system.

B. IMAGE TO WORLD COORDINATE SYSTEM

To transform the information from image coordinates to 2D world coordinates, we modeled our camera projection with the Tsai camera model [40]. With this model, we can get back projection lines that describe a corresponding location in the 2D world coordinate system of each pixel. When more than three image pixels and their corresponding 2D world coordinates are given, the parameters of the Tsai camera model can be found. As in the lidar sensor calibration, we use the specially designed structure to obtain corresponding points between an image and the 2D world coordinate systems.

VII. OBJECT DETECTION

In order to find which measurements are from objects of interest, our algorithm clusters input point measurements and classifies the resulting clusters into an object or a background. In the clustering step, the measurement set at time t, $\mathbf{Z}^t$, is partitioned into $\mathbf{C}_k^t = \{z_i^t | i \in \mathbf{I}_k^t\}, k = 1, \ldots, K^t$, which satisfies the following conditions:

1) $\mathbf{I}_k^t$ is a set of indices of measurements of $\mathbf{C}_k^t$;

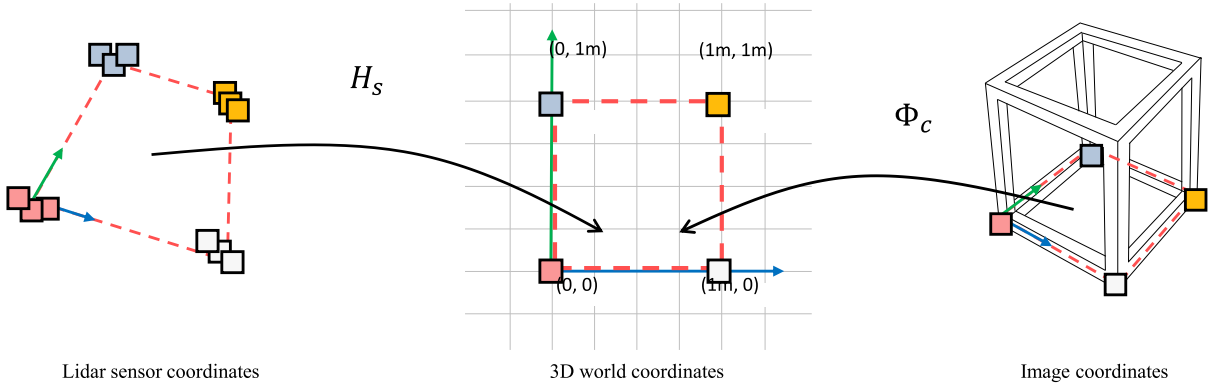


FIGURE 4. Overview on calibration for coordinate transformation.

2) $\cup_{k=1}^{K^t} \mathbf{C}_k^t = \mathbf{Z}^t$ and $\mathbf{C}_l^t \cap \mathbf{C}_m^t = \Phi$ for $l \neq m$; and
3) $\|x_j - x_i\| \leq \gamma_c, \forall i, j \in \mathbf{I}_k^t$;

where γ_c is the maximum allowable distance between the measurements in the same cluster. Then, cluster $\mathbf{C}_k^t$ is classified into an object or background according to the equation defined by

$$f(\mathbf{C}_k^t) = \begin{cases} 1, & \text{if } \|x_i - \bar{\mathbf{c}}_k^t\| \leq r_{max}, \quad \forall i \in \mathbf{I}_k, \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

$$\bar{\mathbf{c}}_k^t = \frac{1}{|\mathbf{C}_k|} \sum_{i \in \mathbf{I}_k} x_i, \tag{7}$$

where $|\cdot|$ is the cardinality of a set.

VIII. TEMPORAL ASSOCIATION

Our algorithm incrementally matches the measurement clusters between consecutive scans to produce an instant tracking result. Most of the existing incremental data association methods [20], [21] have been proposed to track rigid targets that move smoothly through consecutive scans. Our targets, however, are non-rigid and move abruptly, so that ordinary linear or non-linear filters do not contribute much compared to their computational load. Thus, we adopted the Hungarian method [41] to find an optimal match between the previous trajectories and the current measurement clusters with simply defined similarities between them. Let $\mathbf{C}_\mathbf{z}^{\mathbf{t_1}}$ be the set of measurement clusters at time t_1, and let $\mathbf{T}^{t_2}$ be the set of trajectories that have their last location at time t_2. Then, the matching similarity between $c_i \in \mathbf{C}_\mathbf{z}^{\mathbf{t_1}}$ and $\mathcal{T}_j \in \mathbf{T}^{t_2}$ is defined by

$$\lambda_{ij} = \frac{\|\hat{\mathcal{T}}_j^t - \bar{c}_i\|_2}{\Delta_t \times v_{max}} = \frac{\|\mathcal{T}_j^{t-1} + \Delta_t \times v_j^{t-1} - \bar{c}_i\|_2}{\Delta_t \times v_{max}}, \tag{8}$$

where $\bar{c}_i$ is the centroid of c_i, v_{max} is the maximum distance that performers can reach during one scanning interval, $v_j^{t-1} = \mathcal{T}_j^{t-1} - \mathcal{T}_j^{t-2}$, and $\Delta_t = t_1 - t_2$. When extracted motion information from an image camera available, we validate each matching between candidate trajectories and measurement clusters. That is, if the matching between a measurement cluster c_i and a trajectory $\mathcal{T}_j$ contradicts the motion information, we set λ_{ij} to zero.

A. MOTION PRIOR

A lidar sensor is useful to know where an obstacle exists, but it is difficult to extract motion information about how the obstacle moves with a lidar sensor. This is because there is no distinguishable information for each point measurement from lidar sensor. To solve this difficulty, we used an image sensor to extract motion information. The extracted motion information is used to validate the result of temporal association. We first find out the corresponding region to the location of objects detected by a lidar sensor in the image area through the coordinate transformation with calibration information. Then, an optical flow algorithm is applied to the region to extract inter-frame motion of the objects. Since a lidar sensor has a planar scan area, when it is moved to the image, it becomes a very narrow area, which is difficult to extract the optical flow. To resolve this, we assume that each object is at least one meter tall. However, as aforementioned, it is difficult to extract the motion prior if the stage lighting condition is very though. Therefore, the motion prior is used only when the illumination condition is moderate.

IX. RECONSTRUCTION

After temporal association, we can define $\mathbf{Z}_k^t$, the measurements of the k_{th} target at time t. We then reconstruct the trajectory of the target by adopting the method described in [2]. For all t that has non-empty $\mathbf{Z}_k^t$, the estimated location of the k_{th} target is the centroid of $\mathbf{Z}_k^t$:

$$\hat{x}_k^t = \frac{1}{|\mathbf{Z}_k^t|} \sum_{i \in \mathbf{I}_k^t} x_i. \tag{9}$$

If there is some $t' \in [\tau_k^s, \tau_k^e]$ that does not have any measurements for the k_{th} target, i.e., $\mathbf{Z}_k^t = \Phi$, the estimation of the missed location is defined by linear interpolation:

$$\hat{x}_k^t = \hat{x}_k^{t_p} + \frac{t - t_p}{t_f - t} \left(\hat{x}_k^{t_f} - \hat{x}_k^{t_p} \right), \tag{10}$$

where t_p and t_f are the preceding and following times of t that have measurements for the k_{th} target, respectively. The final trajectory of the k_{th} target is obtained by smoothing the estimated locations as

$$x_k^t = \mathcal{F}(\hat{\mathcal{T}}_k, t), \quad t = \tau_k^s, \cdots, \tau_k^e, \tag{11}$$

where $\hat{\mathcal{T}}_k = (\hat{x}_k^{\tau_k^s}, \cdots, \hat{x}_k^{\tau_k^e})$, and $\mathcal{F}(\hat{\mathcal{T}}_k, t)$ is the function returning the smoothed location of $\hat{\mathcal{T}}_k$ at time t. In our experiment, the Savitzky-Golay filter [42] was used as a smoothing function.

X. EXPERIMENTAL RESULTS

The implementation of our method is available at http://bit.ly/multi-target tracking_neohanju. We conducted experiments to examine the robustness of the proposed algorithm and the beneficial effect of using multiple sensors on the tracking performance of our algorithm.

A. DATASET

To our knowledge, there is no available dataset for tracking with multiple single-layer lidar sensors. Thus, we generated a new dataset with two single-layer lidar sensors and two image cameras. We captured the frames in a low lighting condition because that is natural for a stage. The dataset contains two scenarios and each scenario has synchronized frames from scans of multiple sensors. Our dataset also provides calibration information and the ground truth locations of each target.

B. EVALUATION METRICS

To evaluate the performance of our algorithm, we adopted the metrics used in [2] which are precision, recall, MOTA, MOTP, ML, PT and IDS. Please refer to the original literature for the details. For those metrics, a ground truth location is considered to be matched with the closest estimated location when they are closer than ϵ_x, which is set to one meter in normal cases. However, we also conducted additional experiments by setting ϵ_x to two meters because a performer behaving actively can be detected as points spread over two meters.

C. QUANTITATIVE RESULTS

Table 2 shows the evaluation results. We conducted multiple experiments with different combinations of input sensors to examine the effect of using multiple sensors. We highlight the best value of each metric in red. As shown in the table, regardless of what value is used for ϵ_x, using multiple sensors always achieved the best performance with respect to the MOTA metric, which represents a qualitative performance well. Using multiple sensors is also beneficial to precision and recall values.

TABLE 2. Performance evaluation result.

Set	ϵ_x	Sensor IDs	Rec.	Prec.	MT	PT	IDS	MOTA	MOPT
Dance	1.0	1	0.82	0.72	2	1	10	0.47	0.57
		2	0.67	0.68	2	**0**	**0**	0.35	**0.71**
		1+2	**0.99**	**0.99**	**3**	**0**	3	**0.98**	0.60
	2.0	1	0.83	**1.0**	2	1	6	0.82	0.79
		2	0.97	0.98	3	**0**	**2**	0.95	0.67
		1+2	**1.0**	**1.0**	**3**	**0**	3	**0.99**	**0.80**
Rotation	1.0	1	0.76	**0.95**	2	2	11	0.72	**54**
		2	0.72	0.89	1	3	**10**	0.62	48.2
		1+2	**0.87**	**0.95**	**4**	**0**	11	**0.83**	51.6
	2.0	1	0.80	**1.0**	2	2	**10**	0.79	**75.3**
		2	0.81	**1.0**	1	3	**10**	0.80	70.5
		1+2	**0.93**	0.99	**4**	**0**	15	**0.91**	74.2

XI. CONCLUSION

In this paper, we propose a robust online algorithm to track performers on stage with multiple low-cost lidar sensors. Based on the algorithm, we built a simple, cheap and utilizable tracking system that is capable of tracking performers who dance or move abruptly. The performance of the system was evaluated with a newly generated dataset to capture challenging scenarios. According to the result, our tracking system showed a good quantitative performance for those challenging scenarios, especially when the number of sensors was increased. However, out tracking model is quite simple and it does not have any factor about the appearance or dynamic of a target, which is conventionally considered in traditional tracking systems. Therefore, there is room to improve the model of a target's occupancy with its shape and we will consider this issue for our future work.

REFERENCES

[1] M. Byeon, H. Yoo, K. Kim, S. Oh, and J. Y. Choi, "Unified optimization framework for localization and tracking of multiple targets with multiple cameras," *Comput. Vis. Image Understand.*, vol. 166, pp. 51–65, Jan. 2018.

[2] H. Yoo, K. Kim, M. Byeon, Y. Jeon, and J. Choi, "Online scheme for multiple camera multiple target tracking based on multiple hypothesis tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 3, pp. 454–469, Mar. 2017.

[3] K. C. Whitright, D. J. Newman, and K. R. Fasen, "Tracking system," U.S. Patent 5 504 477, Apr. 2, 1996.

[4] K. Lorincz and M. Welsh, "MoteTrack: A robust, decentralized approach to RF-based location tracking," in *Proc. Int. Symp. Location-Context-Awareness*. Springer, 2005, pp. 63–82.

[5] D. Zhang, J. Ma, Q. Chen, and L. M. Ni, "An RF-based system for tracking transceiver-free objects," in *Proc. 5th Annu. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2007, pp. 135–144.

[6] O. Woodman and R. Harle, "Rf-based initialisation for inertial pedestriantracking," in *Proc. Int. Conf. Pervasive Comput.*. Berlin, Germany: Springer, 2009, pp. 238–255.

[7] J. Shackleton, B. VanVoorst, and J. Hesch, "Tracking people with a 360-degree lidar," in *Proc. 7th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug./Sep. 2010, pp. 420–426.

[8] P. Morton, B. Douillard, and J. Underwood, "An evaluation of dynamic object tracking with 3D LIDAR," in *Proc. Australas. Conf. Robot. Automat. (ACRA)*, 2011, pp. 1–10.

[9] J. Yan, D. Chen, H. Myeong, T. Shiratori, and Y. Ma, "Automatic extraction of moving objects from image and LIDAR sequences," in *Proc. 2nd Int. Conf. 3D Vis. (3DV)*, vol. 1, Dec. 2014, pp. 673–680.

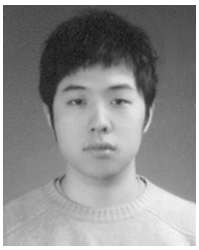
[10] A. Dewan, T. Caselitz, G. D. Tipaldi, and W. Burgard, "Motion-based detection and tracking in 3D LiDAR scans," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 4508–4513.

[11] A. Asvadi, P. Girão, P. Peixoto, and U. Nunes, "3D object tracking using rgb and LIDAR data," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 1255–1260.

[12] L. Huang, S. Chen, J. Zhang, B. Cheng, and M. Liu, "Real-time motion tracking for indoor moving sphere objects with a LiDAR sensor," *Sensors*, vol. 17, no. 9, p. 1932, 2017.

[13] P. Grandjean and A. R. de Saint Vincent, "3-D modeling of indoor scenes by fusion of noisy range and stereo data," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1989, pp. 681–687.

[14] B. R. VanVoorst *et al.*, "Fusion of LIDAR and video cameras to augment medical training and assessment," in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. Intell. Syst. (MFI)*, Sep. 2015, pp. 345–350.
[15] K. Kwak, J.-S. Kim, J. Min, and Y.-W. Park, "Unknown multiple object tracking using 2d lidar and video camera," *Electron. Lett.*, vol. 50, no. 8, pp. 600–602, 2014.
[16] K. O. Arras, O. M. Mozos, and W. Burgard, "Using boosted features for the detection of people in 2D range data," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 3402–3407.
[17] T. Dieterle, F. Particke, L. Patino-Studencki, and J. Thielecke, "Sensor data fusion of LIDAR with stereo RGB-D camera for object tracking," in *Proc. IEEE SENSORS*, Oct./Nov. 2017, pp. 1–3.
[18] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, "A scalable algorithm for tracking an unknown number of targets using multiple sensors," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3478–3493, Jul. 2017.
[19] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, D, J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, 1960.
[20] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE J. Ocean. Eng.*, vol. JOE-8, no. 3, pp. 173–184, Jul. 1983.
[21] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.
[22] R. P. S. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.
[23] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 4, pp. 1224–1245, Oct. 2005.
[24] R. Mahler, "Phd filters of higher order in target number," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 4, 2007.
[25] B. T. Vo, B. N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3553–3567, Jul. 2007.
[26] B.-T. Vo, B.-N. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-Bernoulli filter and its implementations," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 409–423, Feb. 2009.
[27] S. Nannuru, M. Coates, M. Rabbat, and S. Blouin, "General solution and approximate implementation of the multisensor multitarget CPHD filter," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4055–4059.
[28] S. Nannuru, S. Blouin, M. Coates, and M. Rabbat, "Multisensor CPHD filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 4, pp. 1834–1854, Aug. 2016.
[29] S. Nagappa and D. E. Clark, "On the ordering of the sensors in the iterated-corrector probability hypothesis density (PHD) filter," in *Proc. 20th Int. Soc. Opt. Photon. Signal Process., Sensor Fusion, Target Recognit.*, vol. 8050, 2011, p. 80500M.
[30] R. Mahler, "Approximate multisensor CPHD and PHD filters," in *Proc. 13th Conf. Inf. Fusion (FUSION)*, Jun. 2010, pp. 1–8.
[31] M. Tobias and A. D. Lanterman, "Multitarget tracking using multiplebistatic range measurements with probability hypothesis densities," in *Proc. 13th Int. Soc. Opt. Photon. Signal Process., Sensor Fusion, Target Recognit.*, vol. 5429, 2004, pp. 296–306.
[32] G. Battistelli, L. Chisci, S. Morrocchi, F. Papi, A. Farina, and A. Graziano, "Robust multisensor multitarget tracker with application to passive multistatic radar tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 4, pp. 3450–3472, Oct. 2012.
[33] C. Fantacci and F. Papi, "Scalable multisensor multitarget tracking using the marginalized δ-GLMB density," *IEEE Signal Process. Lett.*, vol. 23, no. 6, pp. 863–867, Jun. 2016.
[34] W. Nam, P. Dollár, and J. H. Han, "Local decorrelation for improved pedestrian detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 424–432.
[35] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[36] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.
[37] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4696–4704.
[38] M. Byeon, S. Oh, K. Kim, H.-J. Yoo, and J. Y. Choi, "Efficient spatio-temporal data association using multidimensional assignment in multi-camera multi-target tracking," in *Proc. BMVC*, 2015, pp. 1–68.
[39] P. Morton, B. Douillard, and J. Underwood, "Multi-sensor identity tracking with event graphs," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2013, pp. 4742–4748.
[40] R. Y. Tsai, "An efficient and accurate camera calibration technique for 3D machine vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1986, pp. 364–374.
[41] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logistics Quart.*, vol. 2, nos. 1–2, pp. 83–97, Mar. 1955.
[42] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, 1964.



HAANJU YOO was born in Frankfurt, Germany, in 1985. He received the B.S. and Ph.D. degrees in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 2009 and 2016, respectively. He is currently a Research Assistant Professor with Dankook University, Yongin, South Korea. His research interests include adaptive and learning systems, multi-camera multi-target tracking, and object detection in crowded scenes.

HYEUN JUN MOON received the B.S. degree from the Department of Architecture, Hanyang University, Seoul, South Korea, in 1991, and the Ph.D. degree from the Department of Architecture, Georgia Tech, USA, in 1995. He is currently an Associate Professor with the Department of Architecture, Dankook University, Yongin, South Korea. His research interests include improving indoor air quality of buildings (VOC and micro Water), building energy conservation techniques, airflow distribution analysis, ventilation system development and performance evaluation, heat and condensation, airflow distribution analysis, and computer simulation. He received the Research Fellowship from the Oak Ridge National Laboratory, USA, in 2005.

SEUNG-HOON KIM received the B.S. and M.S. degrees from the Department of Computer Science, Inha University, Incheon, South Korea, in 1985 and 1989, respectively, and the Ph.D. degree from the Department of Computer Science and Engineering, POSTECH, Pohang, South Korea, in 1998. He is currently an Associate Professor with the Department of Architecture, Dankook University, Yongin, South Korea. His research interests include ad hoc networks, wireless sensor networks, multimedia communication networks, routing and topology of high-speed communication networks, and network protocol for distributed services.

SANG-IL CHOI received the B.S. degree from the Division of Electronic Engineering, Sogang University, South Korea, in 2005, and the Ph.D. degree from the School of Electrical Engineering and Computer Science, Seoul National University, South Korea, in 2010. He was a Postdoctoral Researcher with the BK21 Information Technology, Seoul National University, in 2010, and with the Computer Science Department, Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, until 2011. He is currently an Associate Professor with the Department of Computer Science and Engineering, Dankook University, South Korea. His research interests include pattern recognition, machine learning, computer vision, and their applications.